

Hy-dro-gen

Hyphenation patterns from [github:typst/hypher](https://github.com/typst/hypher)

```
#import "@preview/hy-dro-gen:0.1.0" as hy
```

<code>#hy.exists("en")</code>	<code>true</code>
<code>#hy.exists("xz")</code>	<code>false</code>
<code>#hy.syllables("hydrogen")</code>	<code>("hy", "dro", "gen")</code>
<code>#hy.syllables("hydrogène")</code>	<code>("hy", "drogène")</code>
<code>#hy.syllables("hydrogène", lang: "fr")</code>	<code>("hy", "dro", "gène")</code>
<code>#hy.syllables("υδρογόνο", lang: "el")</code>	<code>("υ", "δρο", "γό", "νο")</code>
<code>#hy.syllables("hydrogène", lang: "xz")</code>	<code>panic: Invalid language</code>
<code>#hy.syllables("hydrogène", lang: "xz", fallback: <code>auto</code>)</code>	<code>("hydrogène",)</code>
<code>#hy.syllables("hydrogène", lang: "xz", fallback: "fr")</code>	<code>("hy", "dro", "gène")</code>

The following languages are supported:

Code	Language		Code	Language
<code>"af"</code>	Afrikaans		<code>"sq"</code>	Albanian
<code>"be"</code>	Belarusian		<code>"bg"</code>	Bulgarian
<code>"ca"</code>	Catalan		<code>"hr"</code>	Croatian
<code>"cs"</code>	Czech		<code>"da"</code>	Danish
<code>"nl"</code>	Dutch		<code>"en"</code>	English
<code>"et"</code>	Estonian		<code>"fi"</code>	Finnish
<code>"fr"</code>	French		<code>"ka"</code>	Georgian
<code>"de"</code>	German		<code>"el"</code>	Greek
<code>"hu"</code>	Hungarian		<code>"is"</code>	Icelandic
<code>"it"</code>	Italian		<code>"ku"</code>	Kurmanji
<code>"la"</code>	Latin		<code>"lt"</code>	Lithuanian
<code>"mn"</code>	Mongolian		<code>"no"</code>	Norwegian
<code>"nb"</code>	Norwegian		<code>"nn"</code>	Norwegian
<code>"pl"</code>	Polish		<code>"pt"</code>	Portuguese
<code>"ru"</code>	Russian		<code>"sr"</code>	Serbian
<code>"sk"</code>	Slovak		<code>"sl"</code>	Slovenian
<code>"es"</code>	Spanish		<code>"sv"</code>	Swedish
<code>"tr"</code>	Turkish		<code>"tk"</code>	Turkmen
<code>"uk"</code>	Ukrainian			

The above mapping is available via the static dictionary `#hy.languages`.

- `exists()`
- `syllables()`

Variables

- `languages`

exists

Check if a code corresponds to a language that has registered patterns.

See the list of officially supported languages at [github:typst/hypher](https://github.com/typst/hypher)

If this function returns `true`, then an invocation of `syllables` with this language is guaranteed to not raise a “Invalid language” failure.

Parameters

```
exists(lang: str) -> bool
```

lang `str`

2-letter language code.

syllables

Splits a word into syllables according to hyphenation patterns.

Parameters

```
syllables(
  word: string,
  lang: str,
  fallback: str | auto | none
) -> (..string,)
```

word `string`

Word to split.

lang `str`

Language 2-letter code.

Default: `"en"`

fallback `str` or `auto` or `none`

Determines the behavior in case `lang` is unsupported

- `none`: panics with “Invalid language”
- `auto`: the word is not split at all
- valid 2-letter code: use that instead

Default: `none`

languages `dictionary`

Dictionary of supported codes and languages, in the format:

```
(en: "English", fr: "French", ...)
```

This dictionary is expected but not guaranteed to be in sync with `exists`, because they are fetched through different means. (`exists` queries the actual WASM module, while `languages` is generated automatically from the source code of `hypher`). If they are out of sync, `exists` is the authority for which languages are actually supported by `syllables`.